the prior art of record. Applicants respectfully request the issuance of an advisory action under MPEP 706.07(f).

REMARKS

Objections to and Rejections of the Claims

Claims 1-72 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over US Patent No. 5,485,573 (hereinafter "the Tandon reference") in view of US Patent Number 6,195,760 (hereinafter "the Chung reference") and US Patent No. 4,164,017 (hereinafter "the Randell reference").

In order for a rejection to be proper under 35 U.S. C. § 103(a), the references combined must teach or suggest all the claim limitations. MPEP 2143.03. "All words must be considered in judging the patentability of [the] claim against the prior art." MPEP 2143.03.

Claim 1 recites, *inter alia*, "preserving in place the state of a first set of system resources after the failure occurs in the computer system."

None of the cited references teach or suggest this limitation of claim 1. The Examiner concedes that neither the Tandon reference nor the Chung reference teach this limitation of Claim 1. *See* Final Office Action, page 3. The Tandon reference, in fact, directly teaches away from this limitation by teaching the copying of memory contents to dump files. The Tandon reference recites "the data base management system which detected the error saves to a file the contents of memory which it has allocated." (Tandon, col. 2, lines 41-43, hereinafter references will be made using col:line-line notation, e.g. 2:41-43).

The Chung reference similarly teaches copying the state of a primary application periodically, and using that copied state information to restore the backup copies of the primary application after the primary application failed. The Chung reference recites "using a checkpointing technique to periodically take snapshots of the running state of the primary application module, and storing such state in a stable storage media." (Chung, 1:49-58). The Chung reference further recites

backup application modules that "periodically receive state updates from the primary application module." (Chung 1:66-67). The Chung reference further recites multiple running copies of an application module wherein "states are synchronized among the multiple copies." (Chung, 2:10-11). In each case, the state information is copied from the primary application module to one or more of the various backup application modules. The state information is not preserved in place after the failure occurs.

The Examiner contends that the Randell reference teaches "preserving the state of each item of information after [an] error occurs". A reading of the Randell reference, however, plainly shows that Randell does not preserve in place the state of the items of information after the failure occurs.

The Randell reference teaches a system including a main store 10 and a cache 12. The main store 10 is the memory used during normal operation of the system to store and manipulate data. (Randell, 3:56-60). The cache 12 is a separate memory area used to store backup copies of selected data from the main store 10 that needs to be saved in order to recover from failures (i.e. data that has been changed by the normal operation of the system). (Randell 4:1-6).

The Randell reference teaches a process for saving the state of each of a plurality of items of information at the beginning of execution of a program block (i.e., before the failure occurs) to a backup or cache 12, and subsequently restoring the state of the plurality of items of information from the cache to the main store 10 if a failure is detected. (Randell 1:23-31). The process taught by the Randell reference is quite simple.

When a program block starts execution, the states of the various items of information as generated prior to entry into the program block (e.g. N=1) are already stored in the main store 10. (Randell, Fig. 2, 3:56-60). When the state of an item of information is changed during execution of the block (i.e. N=1 is changed to N=3; Randell 4:1-3), then the prior state of the item of information (N=1) is copied from the main store 10 to the cache 12. (Randell, Fig. 3, 4:3-6). The prior state of the item of information is not preserved in place in the main store 10. Furthermore, this copying occurs before the failure, not after the failure.

When, after completing execution, a program block fails an acceptance test, the states of the items of information which were modified by the failed program block must be restored, so that the computer system can proceed with other operations, such as processing alternate blocks for the failed primary block. (Randell 3:17-22). This restoration is done simply by copying the backup copy of the item of information from the cache 10 to the main store 12. (Randell 5:22-36, FIG. 9). The states of the information items present in the main store 10 at the time of the failure are not preserved. These states are either discarded (if the item of information was local to the failed block, e.g. variable M: Randell 5:27-28) or else the state is restored to the prior value stored in the cache 10 (if the item of information was not local to the failed block, e.g. variable N: Randell 5:30-31). In neither case is the state of the item of information preserved in place in the main store 12.

Claim 1 additionally recites "diagnosing the failure by analyzing one or more resources from the first set of system resources." None of the cited references teach or suggest this limitation of claim 1. As discussed above, the Randell reference teaches no analysis of the items of information copied from the main store 10 to the cache 12 and restored back from the cache 12 to the main store 10 at all. The system of the Randell reference merely discards or overwrites the items of information and then continues with normal operations.

The Chung reference does not teach analyzing resources either. The system of the Chung resource merely monitors the state of a primary application and initiates fail-over procedures if the primary application fails. (Chung 3:16-22). As with the Randell reference, the Chung reference is focused on restoring normal operations after a failure, not on analyzing one or more resources to diagnose the reasons for the failure.

The Examiner contends that the Tandon reference teaches "diagnosing [the] failure by analyzing one or more resources from the first set of system resource[s]" and cites Tandon 2:3-5 for this contention. However, a prior art reference must be considered in its entirety, i.e. as a whole, including portions that would lead away from the claimed invention. MPEP 2141.02 (final section). The cited passage of the Tandon reference merely states that "necessary data" is saved upon detection of a failure, for later analysis. The passage says nothing about what the "necessary data" is.

A full reading of the entirety of the Tandon reference plainly shows that the "necessary data" discussed in the cited passage of the Tandon reference is the copy of the system resources which is made by the system of the Tandon reference. See, for example, "After notifying the data base management systems operating on the other host processor, the data base management system which detected the error saves to a file the contents of memory which it has allocated. Similarly, upon receipt of notification that an error was detected, a data base management system operating on another host processor saves to a file the contents of memory which it has allocated. The timely notification and storage of trace data will assist in determination of the source of the error." (Tandon 2:40-43) This passage plainly shows that the data saved for later analysis is the data copied to a dump file.

Tandon teaches saving copies of resources to dump files, and analyzing the copies of the resources later on.

"The overall processing of the DBMS 24 in incorporating this invention entails processing data base requests from Application Programs 26, detecting an error condition, broadcasting a message to the other Host Processors 10 in the multi-host environment, and saving to a file the contents of memory allocated to the DBMS." (Tandon 4:20-25).

"At step 68, the contents of memory allocated to the DBMS 24 on the sending Host Processor is saved to a dump file. The dump file is then available for analysis by a system programmer for isolating the source of the error." (Tandon 5:30-33).

"The first operation performed by the Snapshot processing of the DBMS 24 on the receiving Host Processor 10 is to save the contents of memory allocated to the DBMS to a dump file as shown by Step 122. The dump file is then available for analysis by a system programmer for isolating the source of the error." (Tandon 7:40-45).

Each of these passages plainly shows that Tandon teaches analysis of copies of resources, not the resources themselves, preserved in place. Tandon plainly does not teach "diagnosing failure by analyzing one or more resources from the first set of system resources" as claimed in claim 1.

Accordingly, by teaching either analysis of copies of resources, or by teaching no analysis at all, the Tandon, Chung and Randell references teach away from "diagnosing failure by analyzing one or more resources from the first set of system resources", and therefore claim 1 and its dependent claims are not obvious over the cited references. Independent claims 12, 21, 22, 41 and 52, and their dependent claims are similarly not obvious over the cited references.

Conclusion

Applicants submit that the claims are in condition for allowance, which is respectfully requested. Nothing in this document shall act as an admission that any reference cited in the Office Action is prior art. Should the Examiner have any questions or comments, he is invited to call the undersigned Attorney at (949) 567-2300.

Respectfully submitted,

LYON & LYON LLP

Dated: July 5, 2002

By: Donald Daybell

Donald Daybell
Reg. No. 50,877

633 West Fifth Street, Suite 4700
Los Angeles, California 90071-2066
(408) 993-1555